



## Script Language description

### General Syntax

The general style is Pascal:

- x There is no case sensitivity.
- x The program starts with **begin** and stops at **end..**
- x ( ) - brackets are used in mathematical formulas.
- x Strings are enclosed in ' ' ( " " is also working).
- x Decimal separator for floating point numbers is a point: ".".
- x Each command is completed with a semicolon: ";".
- x Available mathematical operations:                    +   -   \*   /   ^   e
- x Available relational operations:                        =   >   <   <>

### Variables & Assignments


Variables are single characters only. The data format of all variables is real.

Predefined variables are: **USER1** and **USER2** as well as **Ch** as result of GetChannel.

**Example for variable usage:**                    **A := A + 1.2;**

### Command Overview

Command	Meaning
ClrScr	Deletes the visible data on the data screen
if ... then	Condition
Execute('filename')	Executes another script <i>./script/filename.scr</i>
FeedPara('t', v)	Sets the parameter <i>t</i> for the feedback to the value <i>v</i>
FileName('name1')	Defines the name of the data storage file as <i>./data/name1.txt</i>
for ... to ...	Loop definition
GetChannel( <i>Ch</i> )	Acquires data from output or input channels of the software
GetFeedPara(v)	Gets the value of the feedback parameter <i>v</i> and stores it in the variable <i>c</i>
Goto <i>marke</i>	Jumps to the position <i>marke</i> defined as Label
GoXY(x,y)	Moves tip to a position inside the current scan range
Move(dir,dist)	Uses the current coarse positioning settings to move sample/ tip
ScanPara(p,v)	Sets the parameter <i>p</i> in the ScanParameter window to the value <i>v</i>
ScanImage	Starts to acquire an image
ScanLine( <i>Nr</i> )	Scan ( <i>Nr-1</i> ) lines
SetChannel( <i>Ch</i> , v)	Sets an output channel number <i>Ch</i> to a value <i>v</i>
SetLaser( )	Switches the laser On or Off
SpectPara(n,v)	Sets parameter <i>n</i> for the spectrum acquisition to the value <i>v</i>

	Supplier:	Anfatee Instruments AG, Melanchthonstr. 28, 08606 Oelsnitz
	Device:	Scanning Probe Controlling System -- SXM-Software
	Document:	<b>Language Description</b>

SpectStart	Starts to acquire and save a spectrum with current settings.
Wait(t)	Waits for a time t in seconds
Writeln('text',...)	Displays text in the Memo

## Commands (alphabetical)

### ClrSrc

deletes all visible data and spectra on the memo.

**Example:** `ClrSrc;`

### if condition then command

Evaluation of conditions. ELSE is not supported.

**Example:**

`if (c < -160) then writeln('Channel X =', c);`

### Execute (scriptfilename)

Calls and executes another script file.

**Parameters:** `scriptfilename`: string

**Example:**

`Execute ('Ref-HOPG');` → calls the script “*./scripts/Ref-HOPG.scr*”.

### FeedPara ('type', value)

Sets parameters for the feedback system.

**Parameters:** `type`: string from list

allowed values: Ref, Ki, Kp, Bias, Flyheight

`value`: real

**Example:**

`FeedPara ('Ki', 100);` → sets Ki in the parameter window to 100

### FileName (YourFileName)

defines a file name for the log file, which collects all data during one script run.

If no file name is defined in the script, the system writes its messages into a file *NoName.txt*. With each run of the script, old data in *NoName.txt* are deleted.

If a file name is defined, the file name given by the user is extended automatically by the system with a number representing current date and time (date-time-string), so that the log file is *YourName\_DateTime.txt*. Thus, the user does not need to rename its file in the script for every run.

**Parameters:** `YourFileName` : string → file is *./data/YourFileName\_DateTime.txt*

**Example:** `FileName ('t3');` → the data are written to *./data/t3\_XXXXXXXX.txt*

### for start counter to stop counter do begin command(s) end;

Loop definition based on an integer counter. The commands `do begin` and `end` are always required, even if there is one program line in between, only.



Supplier: Anfatec Instruments AG, Melanchthonstr. 28, 08606 Oelsnitz  
Device: Scanning Probe Controlling System -- SXM-Software  
Document: **Language Description**

**Example** (measures 20 points along a line in x-direction):

```
X:=100;  
for i:=1 to 20 do begin  
  Move('CX',X); end;
```

### GetFeedPara ('type')

Sets parameters for the feedback system.

**Parameters:** **type:** string from list of values: Ref, Ki, Kp, Bias, Flyheight  
**result:** real number stored in the variable c

**Example:**

```
GetFeedPara('Ki'); Writeln(c);  
→ writes the value of Ki on the screen
```

### GetChannel(channelnumber)

Acquires the data of one A/D input channel and stores them in the variable “c”.

**Parameters:** **channelnumber:** integer

For the A/D-Channels, the channel number is assigned as Port in the Scale-Window as Port. Here, “32” stands for the AD1-channel (=> ”T-B”). To get data from the D/A-channels, one choses negative channel numbers. E.g, “0” equals the channel “Z” and “-1” the channel “Bias”. Channel “9” is used for the digital I/O's.

**Example:** **GetChannel (32) ;** → reads the input data of the channel AD1

### Goto marke

Allows to jump to a *marke*. It requires:

- the definition of a label with a **LABEL** statement
- a **Goto** command followed by the labels name
- the label followed by a colon

Labels always start with a character..

**Example:**

```
LABEL foo, exit;  
begin  
foo:  
  GetChannel(32);  
  if c > 1000 then Goto foo;  
exit: writeln('Done. ');  
end.
```

### GoXY (x\_rel, y\_rel)

Moves the tip to the relative positions “x\_rel” and “y\_rel” inside the currently given scan range. The upper left corner of the image equals (0,0), while the lower right corner is (1,1).

**Parameters:** **x\_rel, y\_rel:** real numbers between 0 and 1

**Example:** **GoXY (0.5, 0.5) ;** → moves the tip to image center  
**GoXY (0.1, 0.1) ;** → moves tip to a point close to the upper left corner



Supplier: Anfatec Instruments AG, Melanchthonstr. 28, 08606 Oelsnitz  
Device: Scanning Probe Controlling System -- SXM-Software  
Document: **Language Description**

### Move ('direction', distance)

Allows to operate the coarse positioning system, to set the center position for whole images and to address the functions behind the approach button and retract button in the parameter window. Latter are addressed by their number in the item list, that appears with the right mouse button.

**Parameters:** **direction** : string from list:  
allowed string values: cx, cy, cz, x, y, retract, approach  
**distance** : real → value in physical units or steps

**Examples:** **Move ('CX', 10);** → moves the stepper 10 steps in x-direction  
**Move ('Y', -1.5);** → sets the center position to -1.5  $\mu\text{m}$   
**Move ('Approach', 2);** → starts automated approach  
**Move ('Retract', 0);** → retracts the piezo, only

### ScanPara ('type', value)

allows to set scan parameters in the window "Parameter". With the type 'Autosave', the knob 'Autosave' for the image acquisition is toggled.

**Parameters:** **type** : string from list:  
allowed string values: range, speed, pixel, x,y, angle, linenr, autosave  
**value** : real → value in physical units

**Examples:** **ScanPara ('Speed', 1.5);** → line speed set to 1.5 lines/s  
**ScanPara ('X', -0.5);** → image center X set to -0.5  $\mu\text{m}$   
**ScanPara ('Autosave', 1);** → switches automated save on

### ScanImage

Starts the acquisition of an image. When two or more images needs to be acquired, one has to repeat this command.

**Example:** **ScanImage;** → starts to acquire a whole image

### ScanLine (Number)

Starts the scan of (Number-1) lines with the current settings. If the scan mode is "2D", it starts to scan a number of lines in an image at the current scan position. If the scan mode is "1D" and "cont. on restart" is off, it scans the 1<sup>st</sup> line of an image. In "1D" with "cont. on restart" off, it scans the same line (number-1) times at the current position.

**Example:** **StartLine(10);** → starts to acquire 9 lines with the current settings

### SetLaser(value)

Switches the laser on or off.

**Example:** **SetLaser(0); Wait(1); SetLaser(1);**  
→ switches the laser off for 1 second

### SpectPara (number, value)

allows to set scan parameters in the window "spectroscopy". With the type 'Autosave', the knob



